

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave Blank)		2. REPORT DATE 27 January 2000		3. REPORT TYPE AND DATES COVERED Final - 28 Apr 1999 - 28 Jan 2000
4. TITLE AND SUBTITLE Final Report - Innovative Visual Modeling Environment for Turbine Engine Reverse Modeling Aid Program (TERMAP)			5. FUNDING NUMBERS Contract No.: F33615-99-C-2929	
6. AUTHOR(S) Leigh E. Atkins, SRS Technologies Charles P. DePlachett, SRS Technologies				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) SRS Technologies 500 Discovery Drive Huntsville, AL 35806			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) AFRL/PRTA, Bldg 18 ATTN: Christopher Norden 1950 Fifth Street Wright-Patterson AFB, OH 45433-7251			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) Report developed under SBIR contract for topic AF99-205. There is a need for an innovative Graphical User Interface (GUI), or Visual Modeling Environment, for the Turbine Engine Reverse Modeling Aid Program (TERMAP) which is used by DoD, NASA, and contractors for modeling gas turbine engine propulsion systems. The development of such a user-friendly, innovative GUI was undertaken during this Phase I by SRS Technologies. This Phase I SBIR contract has resulted in a user-friendly, Windows-based GUI which meets and exceeds the performance objectives. This report summarizes the Phase I results and software capabilities. Visual TERMAP can import existing TERMAP input files or create new engine cycle configurations using the drag and drop interface. Visual TERMAP automatically tracks station numbers and TERMAP's Parameter Index Codes (PICs). Compressor and turbine maps can be scaled and plotted. Visual TERMAP also performs trade studies and graphs results using an integrated graphing program. Software deliverables include the Visual TERMAP software program.				
14. SUBJECT TERMS SBIR Report Jet Propulsion Cycle Analysis Graphical User Interface Engine Performance TERMAP			15. NUMBER OF PAGES 18	
17. SECURITY CLASSIFICATION OF REPORT Unclassified			16. PRICE CODE	
18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified		19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified		20. LIMITATION OF ABSTRACT UL

Standard form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

DTIC QUALITY INSPECTED 4

REF D

Final Report

Report Date:
January 27, 2000

Sponsored by:
AFRL/PRTA

Contract No. F33615-99-C-2929

Principal Investigator:
Charles DePlachett

SRS Technologies
500 Discovery Drive
Huntsville, AL 35806
(256) 971-7000

Reporting Period:
April 28, 1999 – January 28, 2000

Effective Date of Contract:
April 28, 1999

Contract Expiration Date:
January 28, 2000

Short Title of Work:
**Innovative Visual Modeling Environment for
Turbine Engine Reverse Modeling Aid Program**

20000201 068

**Innovative Visual Modeling Environment for
Turbine Engine Reverse Modeling Aid Program**

Acknowledgement

SRS Technologies sincerely appreciates the opportunity to perform this Small Business Innovative Research (SBIR) contract for the Air Force Research Laboratory (AFRL) / Propulsion Directorate Turbine Engine Division Engine Integration and Assessment Branch (PRTA). We gratefully acknowledge the professional attitude, courtesy, and technical guidance extended by the Program Manager, Mr. Chris Norden. We also acknowledge the following people for their valuable contributions to this successful research effort: Mr. Greg Bruening, Mr. Nick Kuprowicz, Mr. Mike Gahn, Mr. Glenn Blevins, Mr. Joe Fulford, and Mr. Francisco Rodriguez.

**Innovative Visual Modeling Environment for
Turbine Engine Reverse Modeling Aid Program**

Table of Contents

SECTION	PAGE
1.0 PROJECT SUMMARY	1
2.0 TECHNICAL OBJECTIVES	2
3.0 TECHNICAL APPROACH	2
4.0 PHASE I RESULTS	3
4.1 Task 1 - TERMAP Input Requirements	3
4.2 Task 2 - Engine Cycle Drag & Drop Interface	4
4.3 Task 3 - Component Input Displays.....	5
4.4 Task 4 - Mapping & Scaling Performance Maps	8
4.5 Task 5 - TERMAP PIC Library and Off-Design Window.....	10
4.6 Task 6 - Graphing Package for Displaying Output Data.....	12
4.7 Task 7 - Test & Package TERMAP GUI Demonstrator Software	13
4.8 Task 8 - Phase II Planning.....	13
5.0 CONCLUSIONS & PHASE I DELIVERABLES	14
ATTACHMENT 1 – Phase I Visual TERMAP User’s Manual	

**Innovative Visual Modeling Environment for
Turbine Engine Reverse Modeling Aid Program**

List of Exhibits

EXHIBIT	PAGE
1-1 Phase I Visual TERMAP Software	1
3-1 Phase I Task Schedule	2
4-1 Phase I Object-Oriented Design.....	4
4-2 Phase I Engine Cycle Drag & Drop Interface.....	5
4-3 Inlet Component Properties Window	6
4-4 Split Fan Inlet Component Properties Window.....	6
4-5 Fan Component Properties Window	6
4-6 Compressor Component Properties Window	6
4-7 Splitter Component Properties Window	6
4-8 Duct Component Properties Window	7
4-9 Burner Component Properties Window	7
4-10 Reheat Component Properties Window	7
4-11 Turbine Component Properties Window.....	7
4-12 Mixer Component Properties Window	7
4-13 Bleed Reentry Component Properties Window.....	7
4-14 Nozzle Component Properties Window	8
4-15 Shaft Component Configuration Window	8
4-16 Compressor/Fan Map Listing in the Map Library Window	8
4-17 Turbine Map Listing in the Map Library Window.....	8
4-18 Fan Performance Map.....	8
4-19 Turbine Performance Map	9
4-20 Graph Properties Window.....	9
4-21 Compressor Map Wizard – Design Point Setup Window	9
4-22 Compressor Map Wizard – Intercept / Beta Lines Setup Window.....	10
4-23 Compressor Map Wizard – Scaled Design Point Setup Window.....	10
4-24 Parametric Index Code System Library.....	11
4-25 Parametric Case Setup Window – Basic Parameters	11
4-26 Parametric Case Setup Window – Advanced Parameters	12
4-27 Interface to Pinnacle Publishing Graphics Server	12
4-28 Graphing Design and Off-Design Cases in Visual TERMAP.....	13

1.0 — PROJECT SUMMARY

There is a need for an innovative Graphical User Interface (GUI), or Visual Modeling Environment, for the Turbine Engine Reverse Modeling Aid Program (TERMAP) which is used by DoD, NASA, and contractors for modeling gas turbine engine propulsion systems. Since TERMAP is based on cumbersome text inputs and outputs, it is difficult to learn and inefficient to use. The time required for new users to become competent in running TERMAP is extensive and costly.

The development of such a user-friendly, innovative GUI was undertaken during this Phase I by SRS Technologies. This Phase I Small Business Innovative Research (SBIR) contract for an Innovative Visual Modeling Environment for TERMAP has resulted in a user-friendly GUI that automates the cumbersome data formatting and processing required by TERMAP,

exercises TERMAP in single or batch mode, automatically reads the TERMAP output file, and graphs the engine performance results. SRS Technologies' Visual TERMAP GUI provides a human-factors-engineered, user-friendly, graphical gas turbine engine modeling environment. **Exhibit 1-1** shows the resulting Phase I software entitled "Visual TERMAP". Deliverables include a working Windows-based Demonstrator Software Program that setups the engine configuration, analyzes the configured simulation, and executes the graphics plotting software package. These processes occur seamlessly and are executed in the background. Visual TERMAP will improve the current modeling techniques for propulsion system performance prediction and allow parametric engine trade studies for optimal designs. This will likewise reduce development times and costs to the Air Force Research Laboratory (AFRL) and DoD for manned and unmanned air vehicle (UAV) applications.

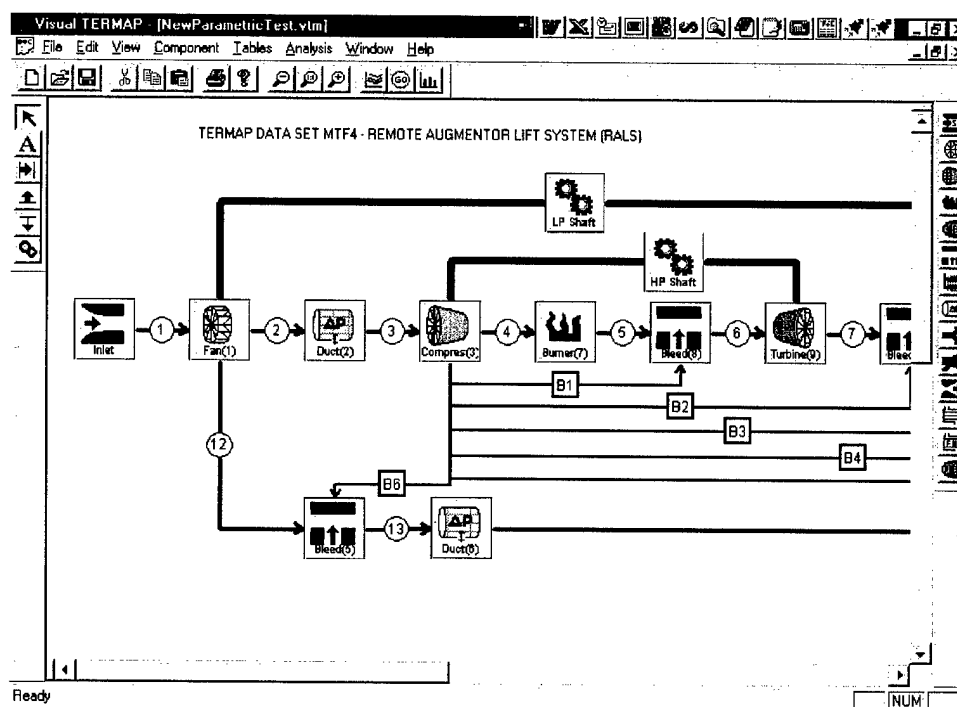


Exhibit 1-1 Phase I Visual TERMAP Software

2.0 — TECHNICAL OBJECTIVES

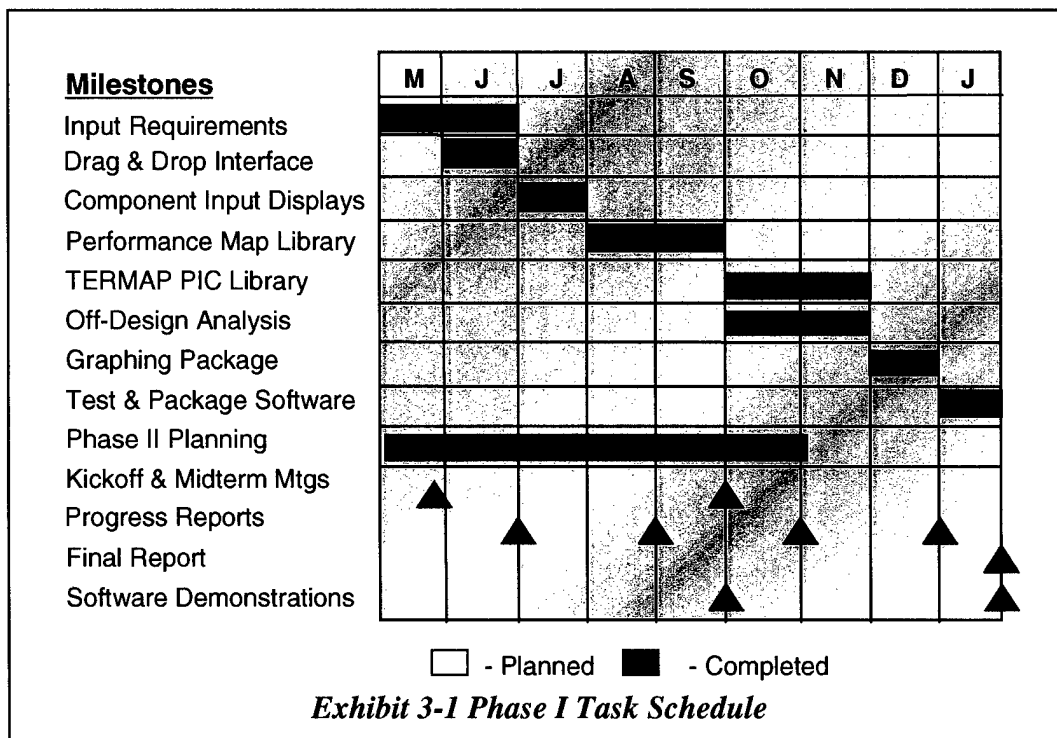
The objectives of this Phase I effort were as follows:

- Identify TERMAP input requirements.
- Develop engine cycle drag and drop interface, which automatically book keeps station and Parameter Index Code (PIC) numbers.
- Design and develop interface for mapping and scaling performance maps.
- Organize PICs into a library to be used in computing design and off-design points and specific outputs.
- Design and develop off-design input window for single and multiple cases.
- Integrate graphing package for displaying output data.

- Test and package Phase I Demonstrator Software.
- Perform Phase II planning for complete Visual TERMAP analysis package.

3.0 — TECHNICAL APPROACH

The overall objective of this Phase I effort was to demonstrate the benefits and feasibility of developing a GUI to make TERMAP a fully capable, user-friendly gas turbine propulsion modeling tool. The approach for accomplishing the Phase I objectives are summarized in **Exhibit 3-1**. These tasks were performed at SRS's Systems Technology Group facility located at 500 Discovery Drive NW, Cummings Research Park, Huntsville, Alabama.



4.0 — PHASE I RESULTS

The Phase I Visual TERMAP software successfully allowed the user to graphically construct engine cycle schematics, analyze basic engine design and off-design conditions, and graphically present the TERMAP results. The following sections describe the end results from each task.

4.1 — Task 1 - Identify TERMAP Input Requirements

During Task 1, the input requirements for exercising TERMAP were identified. These included the following:

- Required files and formats
- TERMAP's Namelist variables
- Engine component order
- Flow station order and numbering
- Engine component inputs/outputs
- Engine operating conditions
- PIC Library variables and indices

Identifying these requirements began with reviewing the TERMAP User's Manual. In addition, we attended an informal TERMAP training session at Wright-Patterson after the Kick-Off Meeting. This training session was very beneficial for two reasons: 1) we were provided a detailed explanation of an example turbofan cycle, and 2) we were provided a version of TERMAP that outputs all cycle variables and integrates well with our GUI. We also obtained useful inputs from TERMAP Users for developing several different areas of the GUI (i.e. PICs, Output Variables, etc...). We received the TERMAP software (executable only) from both the AFRL and Allison, but only the AFRL TERMAP executable was used.

Component Library

Although TERMAP has 19 different types of engine components available, this includes two different splitter components, two different mixer components, and three different nozzle components. Since the type of splitter, mixer, and nozzle can be determined from its position in the engine cycle, it was apparent that the GUI only needs to incorporate one type of splitter, mixer, and nozzle, and depending on where these are located in the cycle, the GUI will determine their exact type. For example, if a mixer is used in an engine cycle, the GUI will determine whether the mixer is used for a bypass or auxiliary flow, and then write the correct type to the TERMAP input file.

Splitter Behavior

For bypass flow from a fan or compressor, a splitter is no longer used. The bypass flow comes directly off the fan or compressor. The GUI inserts the splitter in the TERMAP input file as necessary. Within the GUI, a splitter is only to be used for creating auxiliary flow streams.

Inlet

An Inlet component was created for entering engine operating conditions. It is required to be the first component in an engine cycle, so that the GUI knows where to start the station numbering.

Split Fan Inlet

TERMAP incorporates a component for transferring flow computations to the fan tip. Thus, a Split Fan Inlet was created within the GUI to satisfy this requirement and simplify its use. The new Split Fan Inlet can be used for separating the fan blade flow path into hub and tip paths. It can also be used for other types of advanced cycles where an external flow is being introduced into the cycle.

Importing TERMAP Files

The GUI was designed from the beginning to import existing TERMAP data files. When a file is imported, the FORTRAN namelist variables are read and placed into a duplicated namelist variable structure that mimics the TERMAP common block. This ensures that all data is imported and saved including performance maps, tables, and additional design and off-design cases. It was deemed essential that the user should be able to edit every variable in the engine cycle using the GUI. The most common variables are easily accessible, with access to the advanced variables as well.

Engine Component Parameters

TERMAP uses a combination of input values and performance maps to define each component. For operating conditions, the input performance parameters consist of variables ranging from altitude to flight velocity. Until Visual TERMAP was developed, the user was responsible for entering all input data into a Namelist format. This process was time consuming

and frustrating to the user. To simplify this process, the component inputs and operating conditions were identified and organized in a cohesive manner for the design of user-friendly input screens in Task 3.

4.2 — Task 2- Develop Engine Cycle Drag & Drop Interface

An engine cycle drag and drop interface was developed during Task 2. The Graphical Engine Cycle Analysis Tool™ (GECAT™) developed by SRS Technologies for the NASA Engine Performance Program (NEPP) utilized an object-oriented, drag and drop engine schematic interface. The C++ source code for GECAT was modified to interface with the TERMAP cycle flow path guidelines. The object-oriented design of Visual TERMAP is shown in **Exhibit 4-1**.

As noted in the TERMAP User's Manual, setting up compressor bleed flows are among the most difficult for the user to master. The configuration of bleeds was simplified using a drag and drop approach between the fan, compressor, and duct to the

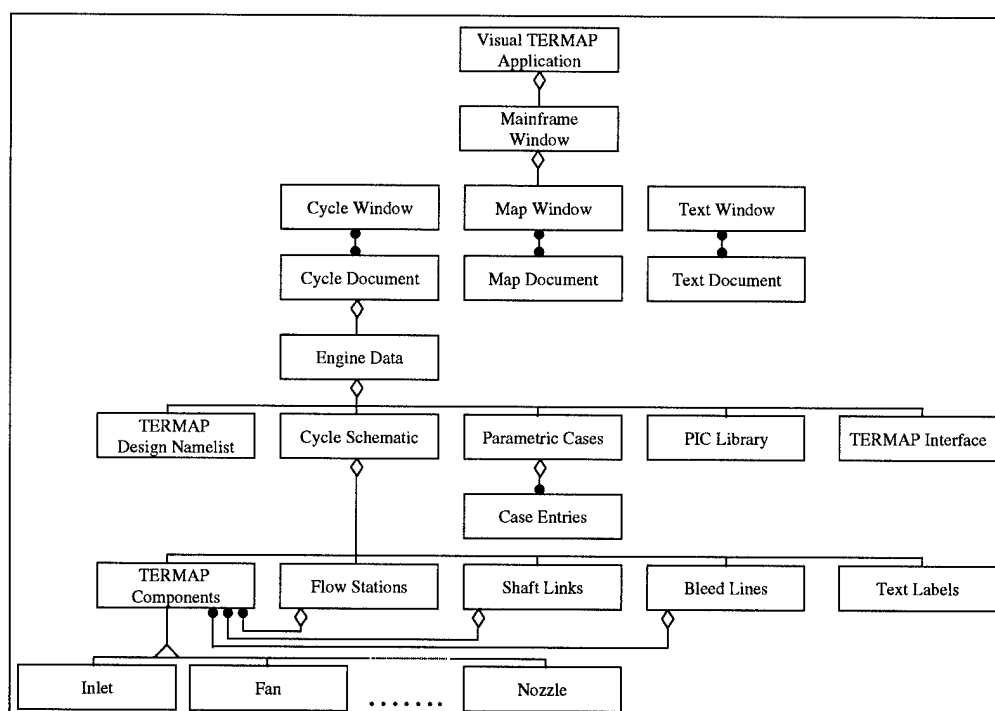


Exhibit 4-1 Phase I Object-Oriented Design

bleed reentry component. Overboard bleeds and externally-supplied bleeds can also be added. The bleed system was revised in TERMAP Version 12, and thus the GUI automatically converts existing data files using the old format to the new Version 12 format.

TERMAP determines how each compressor, fan, and turbine are connected by the user specifying the following type for each: low pressure, intermediate pressure, high pressure, and power turbine (turbine only). The GUI incorporates a Shaft Link window, which the user must specify which type of shaft each component is attached to.

Users are now protected from designing engines that do not adhere to the TERMAP software guidelines. The drag and drop schematic interface automatically catalogs

all station numbers and PIC numbers as new components are inserted into the model. This alleviates the tedious and error-prone book keeping that currently haunts users. **Exhibit 4-2** displays the drag and drop interface developed for Visual TERMAP.

4.3 — Task 3 - Design/Develop Component Input Displays

Task 3 incorporated the TERMAP component input requirements identified in Task 1 into the appropriate component input windows. Beginner and intermediate users of TERMAP tend to have a variety of problems entering correct input values for each component. To increase the user's efficiency, the component input

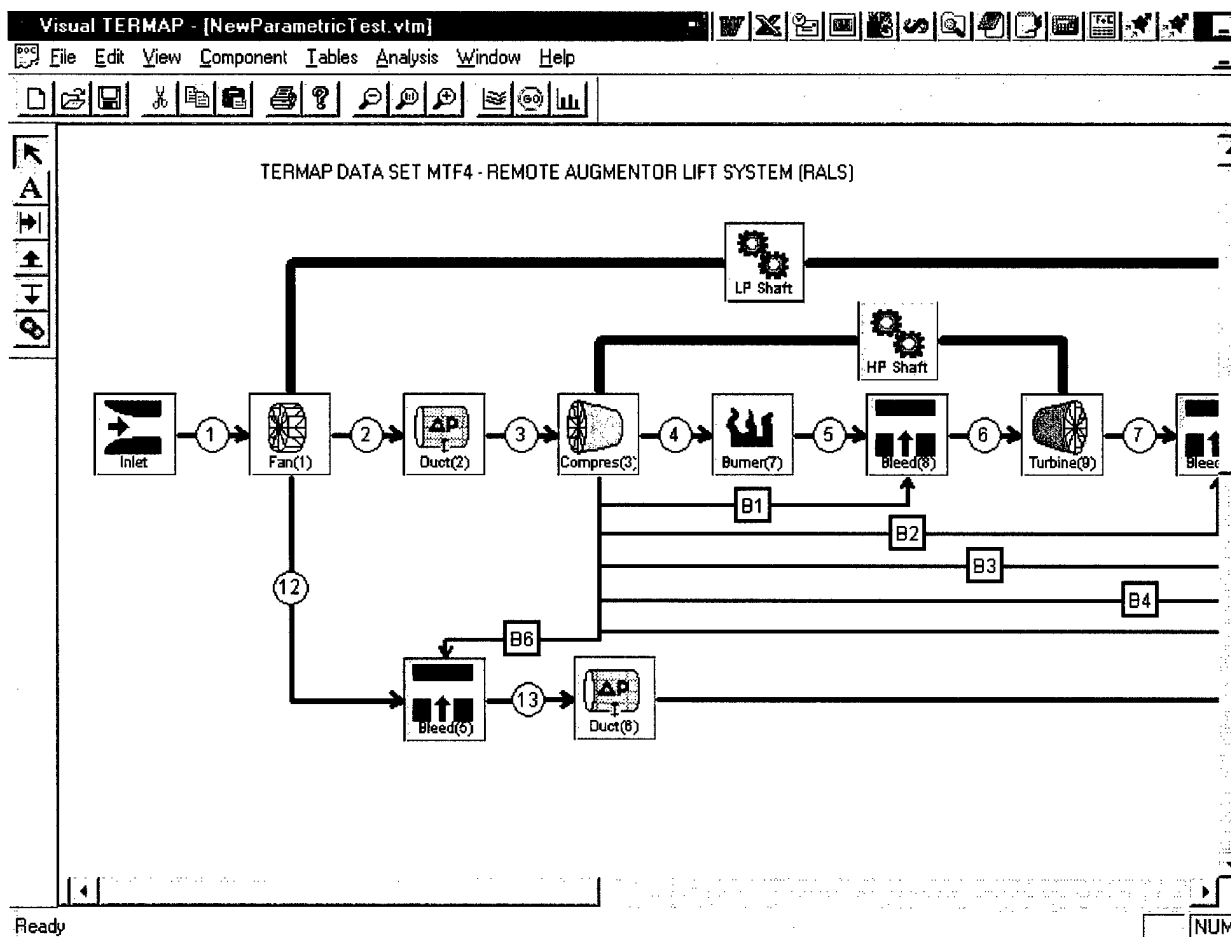


Exhibit 4-2 Phase I Engine Cycle Drag and Drop Interface

requirements were be organized in a cohesive manner to produce user-friendly input screen displays for each component. Tab pages are used within a component window to present the design input variables and the resulting output values in an organized fashion and thereby reducing information overload. Exhibits 4-3 – 4-15 display the component input windows of Visual TERMAP.

Split Inlet(3)

Design | Outputs

Mass Flow Rate: lbm/s

Pressure Recovery:

- > 0 - Recovery
- = 0 - Considered part of total fan
- = -1 - Mil Spec. Recovery
- = -2 - Use Subroutine ERAMX

Ambient Temperature
 Uses Std. Atmospheric Table. Enter Delta T-Ambient =

Defaults Close Apply

Exhibit 4-4 Split Fan Inlet Component Properties Window

Compress(3)

Design | Outputs

Efficiency: %/100

Pressure Ratio:

Map Layer Parameter:

Bypass Ratio:

Power Extracted: hp

Defaults Close Apply

Exhibit 4-6 Compressor Component Properties Window

Inlet

Design | Outputs

Mass Flow Rate: lbm/s

Geopotential Altitude: ft

Flight Velocity:

- < 10 - Mach No.
- ≥ 10 - Knots true air speed
- < 0 - Knots calibrated air speed

Pressure Recovery:

- > 0 - Recovery
- = 0 - Mil Spec. Recovery
- = -1 - Use Subroutine ERAMX

Ambient Temperature
☒ Use Std. Atmospheric Table. Delta T-Ambient =
☐ Ambient Temperature = deg F

Defaults Close Apply

Exhibit 4-3 Inlet Component Properties Window

Fan(1)

Design | Outputs

Efficiency: %/100

Pressure Ratio:

Map Layer Parameter:

Bypass Ratio:

Power Extracted: hp

Defaults Close Apply

Exhibit 4-5 Fan Component Properties Window

Split(2)

Design | Outputs

Bypass Ratio:

Defaults Close Apply

Exhibit 4-7 Splitter Component Properties Window

Duct(15)

Design | Outputs

Pressure Ratio: 0.95

Inlet Area (sq in): 0.2 (Mach No. if < 1)

Defaults Close Apply

Exhibit 4-8 Duct Component Properties Window

Burner(17)

Design | Outputs

Efficiency: 0 %/100

Pressure Ratio: 0.96

Inlet Area (sq in): 0.2 (Mach No. if < 1)

Exit Temperature: 4250 R

Defaults Close Apply

Exhibit 4-9 Burner Component Properties Window

Reheat(116)

Design | Outputs

Pressure Ratio: 1.0 (for cold loss)

Inlet Area (sq in): 0.2 (Mach No. if < 1)

Maximum Hot Core Exit Mach No.: 0

Fraction of Inlet Flow Entering Hot Core: 0 %/100

Reheat Efficiency

☒ Constant Efficiency = 0.81 %/100

☐ Use Default Table with Maximum Efficiency = 0 %/100

Flame Holder Width Factor (in): 0 Burning Length (in): 0

Scale Factor on Table Lookup: 0 Delta on Result: 0

Reheat Flow

Hot Core Max Temperature: 3580 R

Reheat Fuel Flow: 0

Maximum Reheat Fuel (0 to 1.2)

Requested Fuel Flow (> 1.2)

Requested Total Hot Core Fuel-Air Ratio (< 0.0)

Defaults Close Apply

Exhibit 4-10 Reheat Component Properties Window

Turbine(13)

Design | Outputs

Efficiency: 0.02 %/100

Map Layer Parameter: 100

Defaults Close Apply

Exhibit 4-11 Turbine Component Properties Window

Mixer(114)

Design | Outputs

Mixing Gain: 0.08 %/100

Defaults Close Apply

Exhibit 4-12 Mixer Component Properties Window

Bleed(16)

Outputs

Case: 0 (0-based) Total # Cases: 5

Variable	Value	Units

Variable Units: ☒ English ☐ Metric (SI)

Defaults Close Apply

Exhibit 4-13 Bleed Reentry Component Properties Window

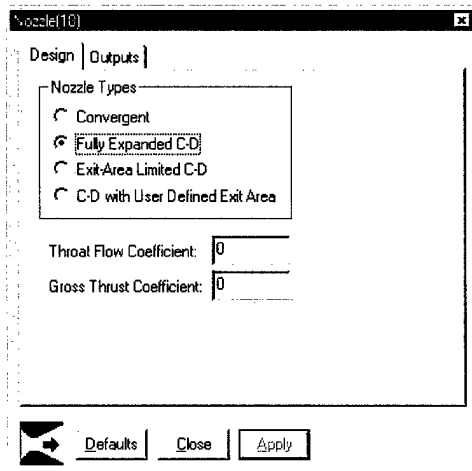


Exhibit 4-14 Nozzle Component Properties Window

4.4 — Task 4 - Design/Develop Interface for Mapping and Scaling Performance Maps

During Task 4, a Map Library was developed to list and display fan, compressor, and turbine performance maps for an engine cycle. From this Library, the user can select and plot a performance map. Different levels of efficiency contour resolution can be used on the fan and compressor performance maps. Design and off-design points can be plotted on any performance map. The user can select whether to scale the map with TERMAP's calculated scale factors or with user defined

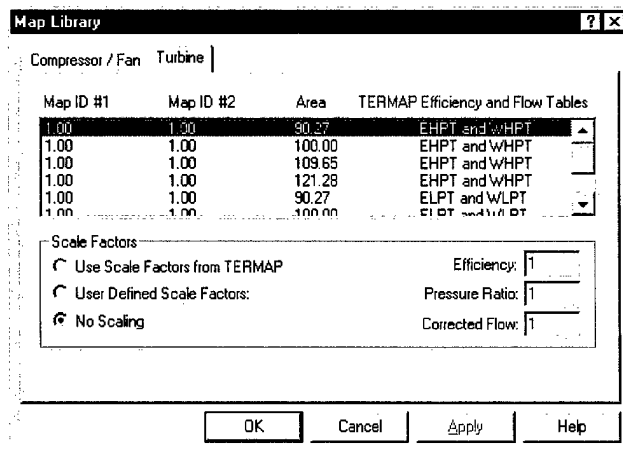


Exhibit 4-17 Turbine Map Listing in the Map Library Window

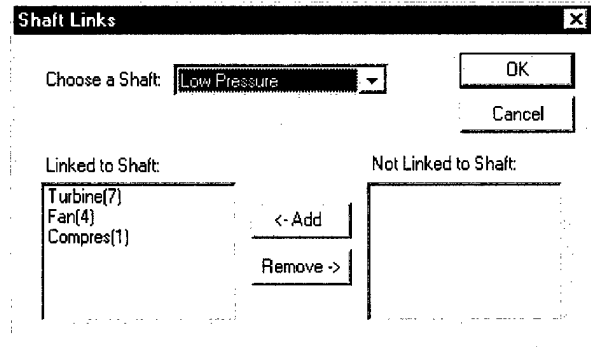


Exhibit 4-15 Shaft Component Configuration Window

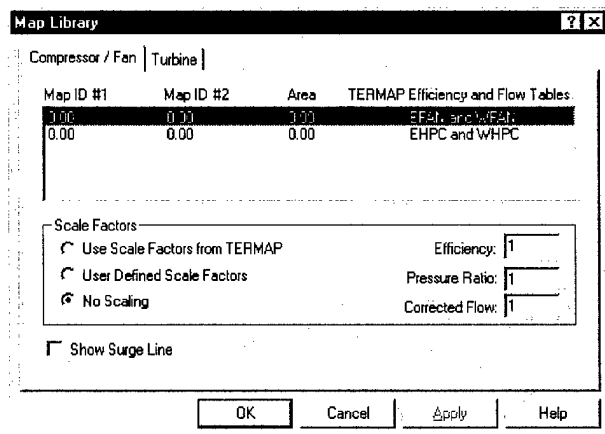


Exhibit 4-16 Compressor / Fan Map Listing in the Map Library Window

scale factors. The performance map can also be left unscaled. The user can change the titles of the x axis, y axis, and legend. The user can zoom in and zoom out on the performance map. Exhibits 4-16 - 4-17 shows the two different tabbed pages of the Map Library. Exhibits 4-18 - 4-19 displays

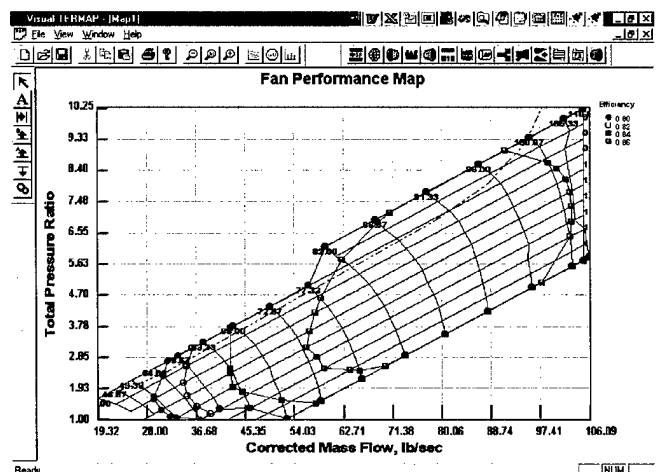


Exhibit 4-18 Fan Performance Map

two performance maps. **Exhibit 4-20** shows the Graph Properties Window used to modify the titles of the axes and to display the design and off-design points on the performance map.

Also during Task 4, a graphical interface was developed for the MAPCOMPR program. This program is used to convert compressor data into a usable map tabulation for TERMAP. The user is responsible for entering all input data into a Namelist format, which can be confusing. Previously, the user was responsible for entering all input data into a complicated Namelist format, which can be confusing. To simplify this process, a Compressor Map Wizard Tool was developed to step the user through the required procedures needed to generate new maps and to scale existing maps. The Wizard Tool creates an input file and executes MAPCOMPR. All of the default values used by MAPCOMPR are included in this Tool. **Exhibit 4-21 - 4-23** displays the windows for the Compressor Map Wizard.

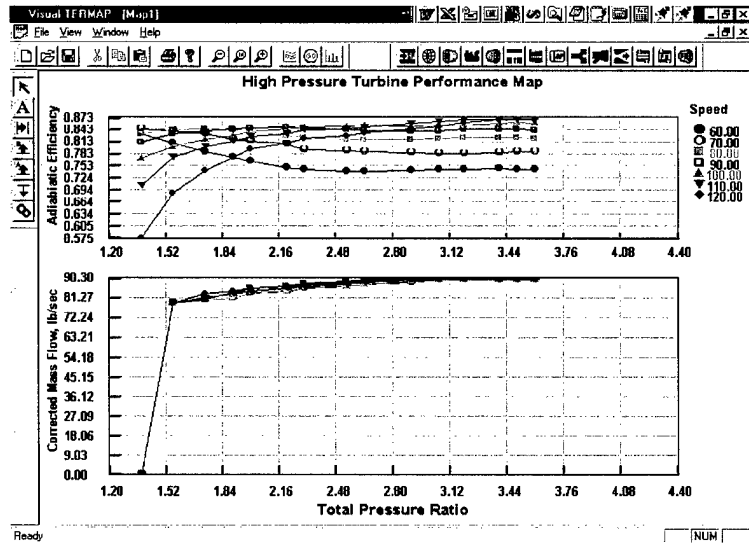


Exhibit 4-19 Turbine Performance Map

Exhibit 4-20 Graph Properties Window

Exhibit 4-21 Compressor Map Wizard – Design Point Setup Window

Compressor Map Wizard - Enter Intercept and Shape of Beta Lines	
<div> <div> Pressure Ratio Intercept of Beta = 1.0 Line </div> <div> Initial Guess for Intercept: <input type="text" value="1"/> </div> <div> Increment Value: <input type="text" value="10"/> </div> <div> Bump Value: <input type="text" value="0.05"/> %/100 </div> <div> Matching Tolerance: <input type="text" value="5e-005"/> %/100 </div> <div> Max Change Per Iteration: <input type="text" value="0.1"/> %/100 </div> <div> Damping Factor: <input type="text" value="0.9"/> </div> <div> Max Iteration Count: <input type="text" value="50"/> </div> </div> <div> <div> Flow Ratio Exponent For Desired Shape of Beta Line </div> <div> Initial Guess for Exponent: <input type="text" value="1"/> </div> <div> Increment Value: <input type="text" value="0"/> </div> <div> Bump Value: <input type="text" value="0.05"/> %/100 </div> <div> Matching Tolerance: <input type="text" value="5e-005"/> %/100 </div> <div> Max Change Per Iteration: <input type="text" value="0.1"/> %/100 </div> <div> Damping Factor: <input type="text" value="0.6"/> </div> <div> Max Iteration Count: <input type="text" value="50"/> </div> </div>	
<div> <input style="border: 1px solid black;" type="button" value=" < Back "/> <input style="border: 1px solid black;" type="button" value=" Next > "/> <input style="border: 1px solid black;" type="button" value=" Cancel "/> </div>	

Exhibit 4-22 Compressor Map Wizard – Intercept / Beta Lines Setup Window

Compressor Map Wizard - Enter New Desired Design Point	
<div> Pressure Ratio: <input type="text" value="4.6"/> </div> <div> Speed: <input type="text" value="100"/> % </div> <div> Number of Speed Lines: <input type="text" value="8"/> </div> <div> Number of Pts per Speed Line: <input type="text" value="4,5,7,7,8,7"/> </div> <div> Number of Speeds per Beta: <input type="text" value="13"/> </div>	<div> <div> Efficiency </div> <div> <input checked="" type="radio"/> Do Not Scale Efficiency </div> <div> <input type="radio"/> Polytropic Efficiency <input type="text" value="0.785"/> </div> </div> <div> Number of Beta Lines: <input type="text" value="13"/> </div> <div> Beta Values: <input type="text" value="7,8,85,9,95,1,0,1,05,1,10,1,15,1,20,1,25,1,3,1,4"/> </div>
<div> <input style="border: 1px solid black;" type="button" value=" < Back "/> <input style="border: 1px solid black;" type="button" value=" Finish "/> <input style="border: 1px solid black;" type="button" value=" Cancel "/> </div>	

Exhibit 4-23 Compressor Map Wizard – Scaled Design Point Setup Window

4.5 — Task 5 - Design/Develop TERMAP PIC Library and Off-Design Window

In Task 5, the Parameter Index Code System (PICs) was organized into a user-accessible categorized library. When the user accesses a variable, which requires a PIC value, then the PIC library is displayed. The PIC library is a special feature of TERMAP used to define cycle parameters

for design, off-design, table lookups, data matching, and output calculations. These PICs are located in a Common Block and are referred to by index number. There are exactly 2,288 different PICs for the user to select for different options during cycle analysis. Visual TERMAP's PIC interface presents a categorized list of the parameters. This augments the user in setting up off-design and data matching, and for viewing specific outputs. The user is able to select a variable name for the PIC instead of trying

to locate the appropriate index number. As mentioned in Task 2, the PIC numbers are automatically adjusted as new components are inserted into the engine model. The user never has to worry about station or PIC numbers changing. The TERMAP GUI writes the correct PIC index number into the input file for the specified operation (i.e. off-design). **Exhibit 4-24** displays Visual TERMAP's Initial Design Case window, which allows the user to modify any input design variable.

Also accomplished in Task 5 was the design and development of the Parametric Cases window. The Parametric Cases window configures design and off-design cases. In order to setup design and off-design cases, the necessary parameters were identified and organized into different two tabbed pages in the Parametric Cases window (Basic and Advanced) to produce a user-friendly input screen. The most common operating conditions and throttling variables were included in the Basic window, while the Advanced window allows any TERMAP engine cycle variable to be changed. The Parametric Cases window uses the PIC Library to set up the proper variables to define the Power Lever Angle (PLA). **Exhibits 4-25 – 4-26** displays the two tabbed pages for the Parametric Cases window.

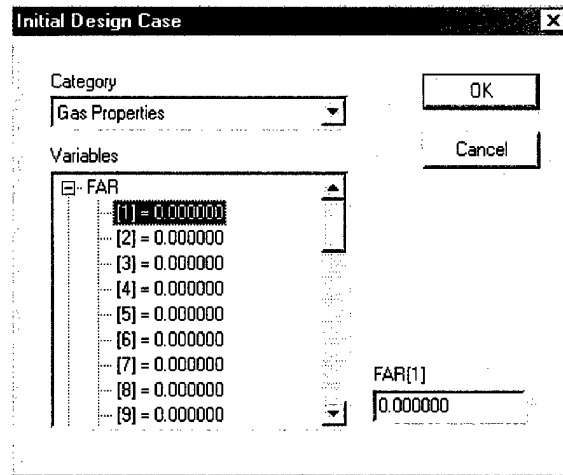
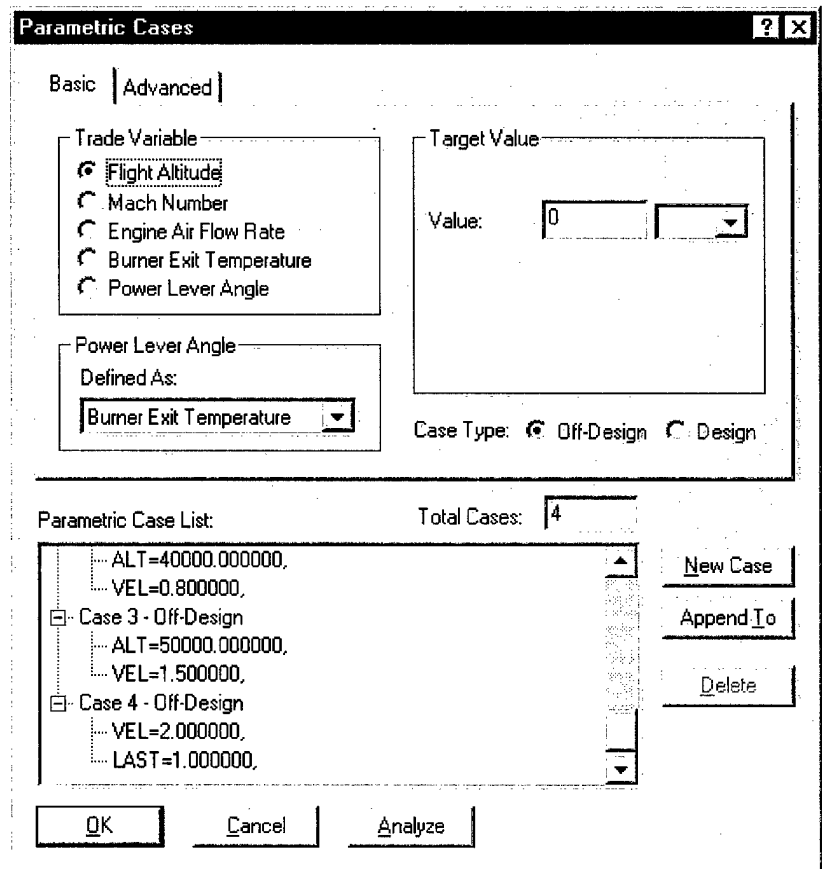


Exhibit 4-24 Initial Design Case Window



**Exhibit 4-25 Parametric Cases Setup Window
– Basic Parameters**

Parametric Cases [?] [X]

Basic | Advanced

Trade Variable

Category:

Variable:

Row Index:

Target Value

Value:

Case Type: ☒ Off-Design ☐ Design

Parametric Case List: Total Cases:

- ALT=40000.000000,
- VEL=0.800000,
- Case 3 - Off-Design
- ALT=50000.000000,
- VEL=1.500000,
- Case 4 - Off-Design
- VEL=2.000000,
- LAST=1.000000,

New Case
Append To
Delete

OK Cancel Analyze

*Exhibit 4-26 Parametric Cases Window
- Advanced Parameters*

4.6 — Task 6 - Integrate Graphing Package for Displaying Output Data

During Task 6, the Pinnacle Publishing Graphics Server SDK was integrated into Visual TERMAP to make graphs and charts within the software (see **Exhibit 27**). Visual TERMAP provides engine performance parameters, component input and output data, station properties, and other pertinent information to the Graphics Server. The Graphics Server uses this data to produce graphs and charts which can be used in other applications (i.e. Microsoft® Word, Microsoft® Power Point, etc...). This capability is very useful when the user is

preparing for demonstrations and briefings. **Exhibit 4-28** shows a graph displaying the gross thrust of several off-design cases from Visual TERMAP.

Graph Data [?] [X]

Graph Type: ☒ 2-D Chart ☐ 3-D Chart

X Data Type:

Y Data Type:

Z Data Type:

X Data Variable:

Y Data Variable:

Z Data Variable:

Units:

Units:

Units:

OK Cancel

Exhibit 4-27 Interface to Pinnacle Publishing Graphics Server

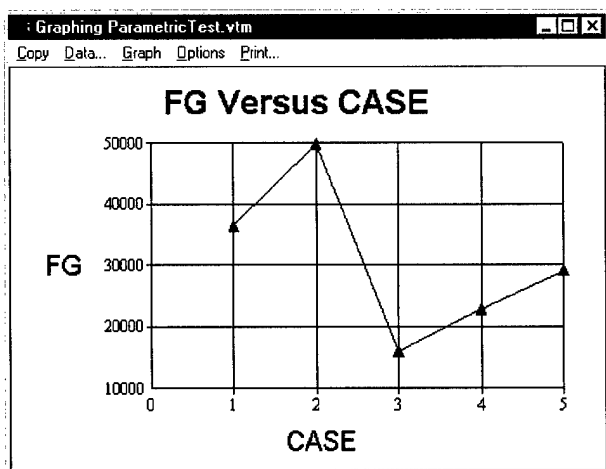


Exhibit 4-28 Graphing Design and Off-Design Cases in Visual TERMAP

4.7 — Task 7 - Test and Package TERMAP GUI Demonstrator Software

In Task 7, Visual TERMAP (Demonstrator Software) was tested to ensure that it could properly import the Turbofan example file provided by AFRL/PRTA. It was also tested to ensure that the Turbofan example could be recreated in Visual TERMAP. Visual TERMAP has the capability to analyze other engine configurations in addition to the example Turbofan engine cycle; however, it has not been tested for these other engine cycles. The Visual TERMAP software program has been packaged on diskette for delivery with the Phase I Final Report. Visual TERMAP demonstrates the time saving-benefits and utility of a user-friendly, cohesive gas turbine engine propulsion modeling tool. The Demonstrator Software sets up the engine configuration, analyzes the configured system, and executes the graphics plotting software package. These processes occur seamlessly and execute in the background. Warning and error messages are displayed as required. The Demonstrator Software features the ability

to display and execute almost any TERMAP-compatible engine model. This ability demonstrates the benefit and utility of graphically visualizing the engine schematic, executing the TERMAP simulation, and graphing engine performance.

Microsoft® Visual C++ was the software development environment used to develop the majority of Visual TERMAP. It allows rapid prototyping of graphic resources. Visual C++ is based on the Microsoft® Foundation Classes (MFC). The MFC library was used extensively to minimize software implementation.

Because of the low cost, availability, and high performance of today's IBM-compatible personal computers (PCs), the Software Demonstrator was developed for a PC running Microsoft® Windows 95, 98, or NT. The performance of a PC has been proven to be more than ample to execute TERMAP models.

4.8 — Task 8 - Phase II Planning

During Phase I, the foundation for the Phase II Visual TERMAP was designed, developed, and tested. The Demonstrator Software was the initial step for beginning Phase II. SRS identified features to be implemented the Phase II. These features (data matching, reverse engine modeling, etc...) were documented along with possible screen displays in the Phase II proposal that was submitted to AFRL/PRTA on October 28, 1999. On January 6, 2000, SRS received notification from AFRL stating that we were not initially selected for Phase II funding. AFRL and SRS are seeking other avenues for potential funding.

5.0 — CONCLUSION & PHASE I DELIVERABLES

The Phase I SBIR contract for the Innovative Visual Modeling Environment for Turbine Engine Reverse Modeling Air Program has resulted in a user-friendly, Windows Based, innovative GUI which meets and exceeds the required performance objectives. The Phase I Visual TERMAP software successfully performs the following:

- Imports existing TERMAP input files (including the Turbofan test file)
- Creates new engine configurations with the engine cycle drag and drop interface (including the Turbofan cycle)
- Automatically tracks station and Parameter Index Code (PIC) numbers.
- Plots and scales component performance maps with operating lines
- Uses a PIC Library for configuring design and off-design points and computing specific outputs.
- Analyzes the configured engine cycle.
- Performs design and off-design trades.
- Executes the graphics plotting software package.

SRS is now prepared to further develop Visual TERMAP to meet the objectives set forth by the Phase II portion of the project. The successful development of the Phase I Visual TERMAP and SRS's prior Windows-Based GUI development experience demonstrate the expertise required to successfully implement the objective of the full-scale GUI required for Phase II.

Report Deliverables:

- Kickoff Briefing
- Interim Progress Reports (4)
- Midterm Briefing
- Final Report

Software Deliverables:

- Visual TERMAP Demonstrator Software Program